Assignment 2

Produce a comparative infographic of TDD, BDD, and FDD methodologies. Illustrate their unique approaches, benefits, and suitability for different software development contexts. Use visuals to enhance understanding.

1. Test Driven Development(TDD)-->

Test Driven Development (TDD) is a software development practice that focuses on creating unit test cases before developing the actual code. It is an iterative approach combining programming, unit test creation, and refactoring.

a. Approach->

In TDD Tests are written before the actual code.

b. Benifits->

1. Early Bug Detection.

2. Improve code quality and maintainability.

3. Adding and testing new functionalities become much easier in the latter stages of development.

c. Suitability->

It is suitable for project which have changing requirement and a need for rapid feedback loops.

2. Behavior Driven Development(BDD)-->

BDD is an approach in software development that emphasizes the behavior of an application for business needs. It was conceived to address issues arising from ill- defined requirement specifications and to align business and QA professionals.

a. Approach->

In BDD Focuses on the behavior of the system from the user's perspective, expressed in natural language specifications

b. Benifits->

1. Collaboration between developers, testers, and business stakeholders.

2. Improved understanding of user requirements

c. Suitability->

It is Ideal for projects with complex business logic and cross-functional teams

3. Future Driven Development(FDD)-->

It is an agile iterative and incremental model that focuses on progressing the features of the developing software. The main motive of feature-driven development is to provide timely updated and working software to the client. In FDD, reporting and progress tracking is necessary at all levels.

a. Approach->

In FDD Divides the project into small, manageable features, with an emphasis on regular builds and client feedback.

b. Benifits->

1. Emphasizes on delivering tangible, working features

2. Clear project progress tracking

3. Suitable for large-scale projects with a focus on feature delivery

c. Suitability->

It is Effective for projects with well-defined requirements and a need for scalability.

Aspect	TDD	BDD	FDD
Approach	Test-first development	Behavior-driven	Feature-based development
Focus	Code functionality	User behavior	Deliverable features
Collaboration	Developer-centric	Cross-functional teams	Team-based approach
Flexibility	High	Medium	Medium
Scalability	Moderate	High	High
Best Suited	Agile environments	Complex requirements	Large-scale projects